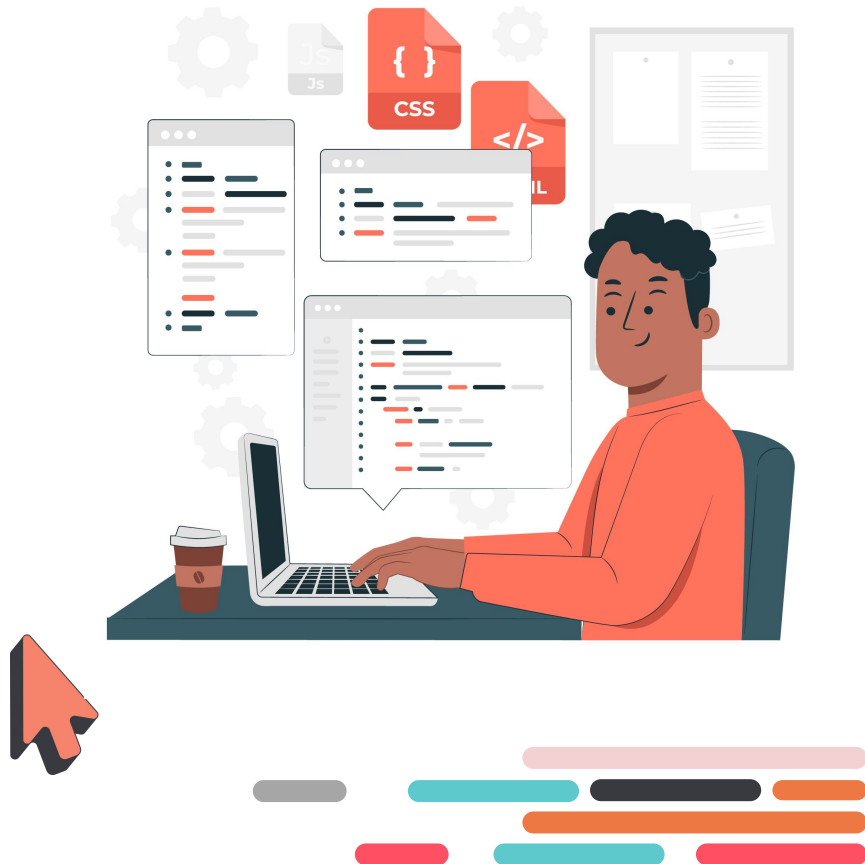




KEMNAKER



CLI, Git & Pengenalan Nodejs





KEMNAKER

sanbercode



Outline

- CLI yang direkomendasikan, Text Editor yang direkomendasikan & Perintah Dasar CLI
- Pengenalan Git & Perintah Git Dasar
- Apa itu Javascript
- Apa itu NodeJS
- Javascript Dasar





KEMNAKER



Software terminal / command line yang direkomendasikan





Windows Powershell

CMder

Git Bash

WSL





KEMNAKER



Text Editor yang direkomendasikan





Visual Studio Code
Sublime Text
Webstorm





Command-Command Dasar CLI





KEMNAKER



1. Menuju folder/direktori tertentu

perintah cd atau kepanjangan dari change directory

// Menuju Desktop

\$ cd Desktop

// Menuju ke direktori parent (sebelumnya)

\$ cd ..





KEMNAKER



2. Mengetahui alamat dari direktori yang sedang dibuka

Terkadang kita butuh tahu sedang dimana posisi terminal kita berjalan. Perintah untuk fungsi ini yaitu `pwd` atau kependekan dari path of working directory

```
$ pwd  
/home/users  
// sedang berada di folder home/users
```





KEMNAKER



3. Mengetahui daftar isi dari sebuah direktori/folder

Kamu bisa gunakan perintah ls atau dir . contohnya sebagai berikut

// di bash

```
$ ls  
Document Desktop Image index.html
```

// di cmd

```
$ dir  
Document Desktop Image index.html
```





KEMNAKER



4. Membuat folder baru

Perintahnya adalah `mkdir` atau singkatan dari `make directory`. contohnya sebagai berikut

// membuat folder baru bernama `new-folder`

```
$ mkdir new-folder
```

// mengecek apakah sudah berhasil buat folder baru

```
$ ls  
new-folder ... ..
```





5. Membuat file baru

Gunakan perintah touch untuk membuat file baru

```
// membuat file index.html  
$ touch index.html
```

```
// mengecek apakah sudah berhasil buat file baru  
$ ls  
index.html ... ..
```





Pengenalan Git





Git

Git adalah sebuah Version Control System yaitu sistem yang mengelola perubahan dari sebuah dokumen, program komputer, website dan kumpulan informasi lain.





NAMA FILE SKRIPSI

- 
-  Skripsi.docx
 -  Skripsi revisi.docx
 -  Skripsi revisi fix.docx
 -  Skripsi revisi final.docx
 -  Skripsi revisi final fix.docx
 -  Skripsi revisi final fix banget.docx



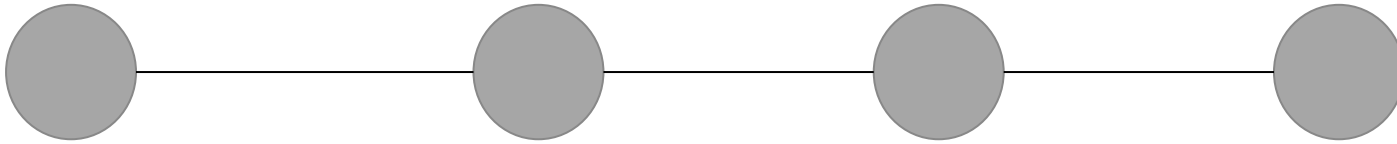
ilustrasi Penggunaan Git

revisi Bab 1

revisi bab 2

revisi bab 3

revisi bab 4





Instalasi Git Ubuntu

```
$ sudo apt update  
$ sudo apt install git  
$ git --version
```





Instalasi Git Mac dan Windows

download dan install dari
<https://git-scm.com/downloads>





Perintah Dasar Git





Git init

git init adalah perintah untuk inisialisasi atau memasang git pada sebuah repository yang sedang terbuka di command prompt atau terminal





KEMNAKER



Git add

git add yaitu perintah git untuk menambahkan perubahan-perubahan didalam direktori project kita yang sebelumnya telah kita lakukan sebuah perubahan.





Git Status

git status adalah perintah dalam git untuk memeriksa status repository, apakah ada file baru, perubahan pada file sudah di “git add ” atau apakah perubahan sudah di commit atau belum.





KEMNAKER



Git Commit

git commit yaitu perintah dari git untuk menyimpan perubahan file yang ada di repository kita, namun file yang tersimpan itu belum terkirim ke remote repository. Perintah commit ini biasanya disertakan dengan -m "Message Commit Anda"





Git Push

git push adalah perintah dalam git untuk mengirimkan perubahan perubahan file yang berhubungan pada repository local di PC/Laptop kita kepada remote repository





Git Pull

git pull adalah perintah dalam git untuk mengambil perubahan terakhir dari repository di git





Remote Repository dan clone repository

Remote Repository adalah tempat dimana kita menyimpan source-code project kita ditempat lain. Biasanya remote repository yang sering digunakan adalah layanan seperti Github, Gitlab, dan yang lainnya.

sedangkan clone repository ialah perintah mengambil project yang ada pada layanan github dan lain-lain ke perangkat yang kita gunakan untuk mengembangkan project tersebut





KEMNAKER



Tutorial Git Sanbercode:

https://www.youtube.com/playlist?list=PL8YCzrSDCR7iZl1LSYyx4h6v6we_qcwZF





KEMNAKER



Sesi Tanya Jawab





Javascript dan Node.js





KEMNAKER



Apa itu Javascript?

Javascript adalah bahasa pemrograman yang digunakan developer untuk membuat halaman web yang interaktif.

https://www.w3schools.com/js/js_history.asp





KEMNAKER



Apa itu Node.js?

Nodejs adalah sebuah tempat dimana kita dapat mengeksekusi kode dalam bahasa Javascript

itu berarti javascript yang biasanya dieksekusi melalui browser, dengan Node.js kita mengeksekusi javascript tanpa browser





KEMNAKER



Menjalankan Javascript

Ada bermacam cara untuk menjalankan kode javascript yang kita buat, di antaranya:

1. Menjalankan javascript di console browser
2. Menjalankan javascript di tools online seperti jsbin
3. Menjalankan javascript dengan nodejs





Instalasi Nodejs

untuk Instalasi Nodejs bisa melalui web resmi nodejs.org





KEMNAKER



Sesi Tanya Jawab





Javascript dasar





Variable

Variable adalah suatu blok data untuk menampung sekumpulan data dengan berbagai tipe data apapun.

untuk penulisan variabel seperti contoh dibawah ini:

```
var nama = "John Doe"  
var nomorUrut = 12
```





Data Type

Data Type atau dalam bahasa indonesia **Tipe Data** adalah sekumpulan informasi yang memiliki nilai dan karakteristik tertentu. Beberapa contoh tipe data pada javascript di antaranya:

1. **Number** : tipe data angka
2. **String** : tipe data berupa text atau kumpulan karakter, biasanya string dibungkus dalam tanda petik ganda (double quote) atau tanda petik tunggal (single quote).
3. **Boolean**: tipe data dengan nilai true atau false





Operator

Operator Javascript terbagi ke dalam beberapa jenis:

- 1. Operator Aritmatika**
- 2. Operator Assignment**
- 3. Operator Perbandingan**
- 4. Operator Kondisional**





KEMNAKER



Operator Aritmatika

Operator yang melibatkan operasi matematika diantaranya:

- Tambah (+)
- Kurang (-)
- Kali (*)
- Bagi (/)
- Modulus (%)





KEMNAKER



Operator Assignment

Operator untuk menyimpan suatu nilai ke dalam suatu variable

contoh:

```
var angka
```

```
angka = 10 // Contoh assignment variable angka dengan nilai 10
```





KEMNAKER



Operator Perbandingan

Operator yang membandingkan suatu nilai dengan nilai yang lain. hasil dari operator perbandingan berupa boolean

1. Equal Operator (==)
2. Not Equal (!=)
3. Strict Equal (===) Selain membandingkan dua nilai nya, strict equal juga membandingkan tipe datanya apakah sama atau tidak
4. Strict not Equal (!==) Kebalikan dari strict equal.
5. Kurang dari & Lebih Dari (<, >, <=, >=)





KEMNAKER



Operator Kondisional

Operator yang mengkombinasikan dua nilai boolean. Terdapat operator AND (&&) dan OR (||).

untuk operator AND (&&) menghasilkan true jika semua hasil kondisional bernilai true

sedangkan untuk operator OR (||) menghasilkan true jika salah satu nilai kondisional bernilai true





KEMNAKER



Sesi Tanya Jawab





String Javascript





Javascript String Methods

.length

mengembalikan panjang atau jumlah karakter pada suatu string.

```
var word = "Javascript is awesome"  
console.log(word.length) // 21
```

.charAt([indeks])

Mengembalikan karakter pada indeks yang diinginkan

```
console.log('i am a string'.charAt(3)); // 'm'
```





Javascript String Methods

`.concat([string])`

Menggabungkan beberapa string dan mengembalikannya menjadi string baru.

```
var string1 = 'good';  
var string2 = 'luck';  
console.log(string1.concat(string2)); // goodluck
```

`.indexOf([string/karakter])`

Mengembalikan indeks dari string/karakter yang dicari, yang pertama kali ditemukan, atau -1 apabila tidak ditemukan.

```
var text = 'dung dung ces!';  
console.log(text.indexOf('dung')); // 0  
console.log(text.indexOf('u')); // 1  
console.log(text.indexOf('jreng')); // -1
```





Javascript String Methods

.substring([indeks awal], [indeks akhir (optional)])

Mengembalikan potongan string mulai dari indeks pada parameter pertama (indeks awal) sampai dengan indeks pada parameter kedua (indeks akhir).|

```
var car1 = 'Lykan Hypersport';  
var car2 = car1.substring(6);  
console.log(car2); // Hypersport
```





KEMNAKER



Javascript String Methods

.toUpperCase()

Mengembalikan string baru dengan semua karakter yang diubah menjadi huruf kapital.

```
var letter = 'This Letter Is For You';  
var upper = letter.toUpperCase();  
console.log(upper); // THIS LETTER IS FOR YOU
```

.toLowerCase()

Mengembalikan string baru dengan semua karakter yang diubah menjadi huruf kecil

```
var letter = 'This Letter Is For You';  
var lower = letter.toLowerCase();  
console.log(lower); // this letter is for you
```





KEMNAKER



Javascript String Methods

.trim()

Mengembalikan string baru yang sudah dihapus karakter *whitespace* (" ") pada awal dan akhir string tersebut.

```
var username    = ' administrator ';  
var newUsername = username.trim();  
console.log(newUsername) // 'administrator'
```

.replace()

mengganti string tertentu menjadi string yang kita inginkan

```
var username = 'administrator';  
var newUsername = username.replace('r', 'R');  
console.log(newUsername) // 'administratoR'
```

.replaceAll()

mengganti string tertentu menjadi string yang kita inginkan

```
var text= 'saya seorang programmer';  
text = text.replaceAll(' ', '-');  
console.log(newUsername) // 'saya-seorang-programmer'
```





KEMNAKER



Mengubah tipe data lain menjadi String atau sebaliknya

String([angka/array])

Fungsi global String() dapat dipanggil kapan saja pada program JavaScript dan akan mengembalikan data dalam tipe data String dari parameter yang diberikan.

```
var int = 12;
var real = 3.45;
var arr = [6, 7, 8];

var strInt = String(int);
var strReal = String(real);
var strArr = String(arr);

console.log(strInt); // '12'
console.log(strReal); // '3.45'
console.log(strArr); // '6,7,8'
```





Mengubah tipe data lain menjadi String atau sebaliknya

.toString()

Mengonversi tipe data lain menjadi string. Bila data tersebut adalah array, setiap nilai akan dituliskan dan dipisah dengan karakter koma.

```
var number = 21;  
console.log(number.toString()); // '21'  
var array = [1,2];  
console.log(array.toString()); // '1,2'
```

Number([String])

Fungsi global Number() mengonversi tipe data string menjadi angka. Data yang diberikan pada parameter harus berupa karakter angka saja, dengan titik (separator) bila angka adalah bilangan desimal. Bila parameter berisi karakter selain angka dan/atau titik, Number() akan mengembalikan **NaN** (*Not a Number*).

```
var number1 = Number("90"); // number1 = 90  
var number2 = Number("1.23"); // number2 = 1.23  
var number3 = Number("4 5"); // number3 = NaN
```





Mengubah tipe data lain menjadi String atau sebaliknya

`parseInt([String])` dan `parseFloat([String])`

Fungsi global `parseInt([String])` dan `parseFloat([String])` mengembalikan angka dari string. Bila angka adalah bilangan desimal maka gunakan `parseFloat()`, bila tidak bilangan dibelakang koma akan diabaikan.

```
var int    = '89';  
var real   = '56.7';  
var strInt_1 = parseInt(int); // strInt_1 = 89  
var strInt_2 = parseInt(real); // strInt_2 = 56  
var strReal_1 = parseFloat(int); // strReal_1 = 89  
var strReal_2 = parseFloat(real); // strReal_2 = 56.7
```





KEMNAKER

sanbercode



Kesimpulan

- Software terminal / command line yang direkomendasikan & Perintah Dasar CLI
- Pengenalan Git & Perintah Git Dasar
- Apa itu Javascript
- Apa itu NodeJS
- Javascript Dasar





Terima Kasih

